Abstract

In this era of growing technology where everything is digitalizing, our university examination section is still using a tradition way for examination form filling and dispatching notices on notice boards all over university. It publishes results on its website which we can say is somewhat modernized but the students are not notified when the results are published. To know if the result is published, a student has to manually check the website every now and then, to check whether the result is published or not. To tackle this problem, we have developed an application which digitalizes all these processes (form filling, exam seat plan notices, and other exam related notices) and which also notifies the students when the result is published. To complete this project, we have used “Android” as the core language and “Firebase” for the database.

Key words: Object Oriented Programming, Firebase

Contents

1. Introduction

1.1 Background

The examination control office in our university still uses a traditional way of form filling for every examination it holds. Students need to manually fill up a form and submit it to the respective offices before deadline. Students also need two hard copies for their passport sized photographs for every examination form they fill up which is somehow traditional and this creates a problem among the student as they might not have their photographs ready with them during the time of form filling. Also when the exam section publishes the result, no one is notified. Student need to check the website everyday manually expecting for their result to get published, which might be quite inconvenient at times. These problems could be solved if the examination had a system which digitalizes the whole form filling process, students can always keep their digital photographs and the system notifies everyone when a result is published.

1.2 Objectives

- Digitalizing the examination form filling process.
- Notifying everyone when the result is published.
- Digitalizing every notices published by exam section.

1.3 Motivation and Significance

The traditional exam form filling process is so hectic and remembering the deadlines and not having a hard copy passport size photograph at the time has always been a problem to the university students. Also running towards notice board just before exam just to find the seat plan is another problem. This has been the main motivation for us to think about making a more interactive system; an examination section management system application that includes online exam form filling, notification system, a user's profile that can be accessed by the specific student who the user is concerned with, the user's online GPA sheet, and digitalizing all the notices published by the examination section. In this way a lot of time can be conserved and the process of traditional exam section processes can be eliminated. And to validate our problem and to further proceed with this project, we conducted a survey, whose result motivated us to develop this application. The below figure shows our survey results:

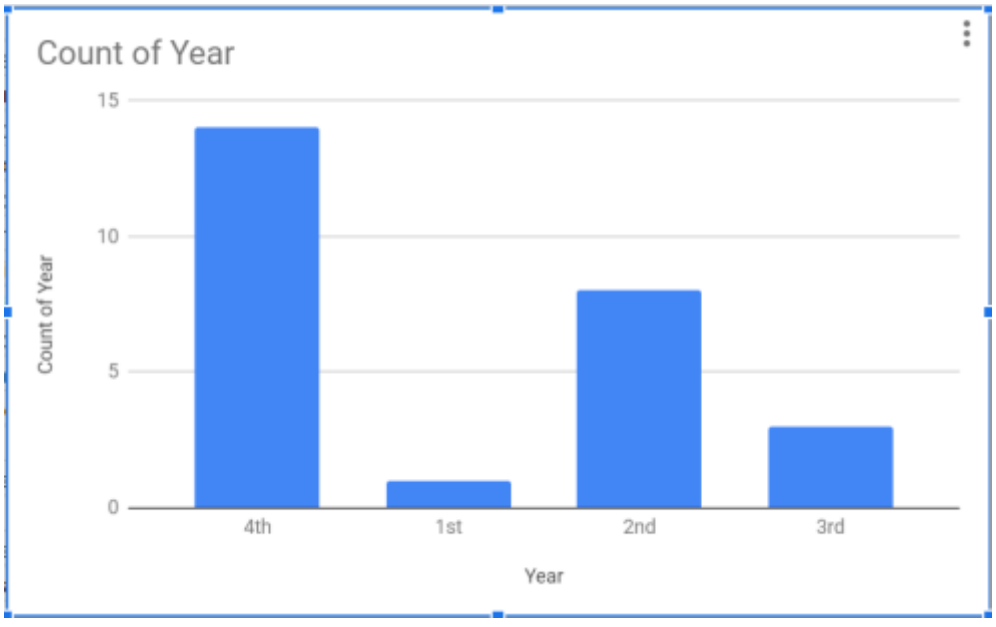


Figure 1: Survey result 1

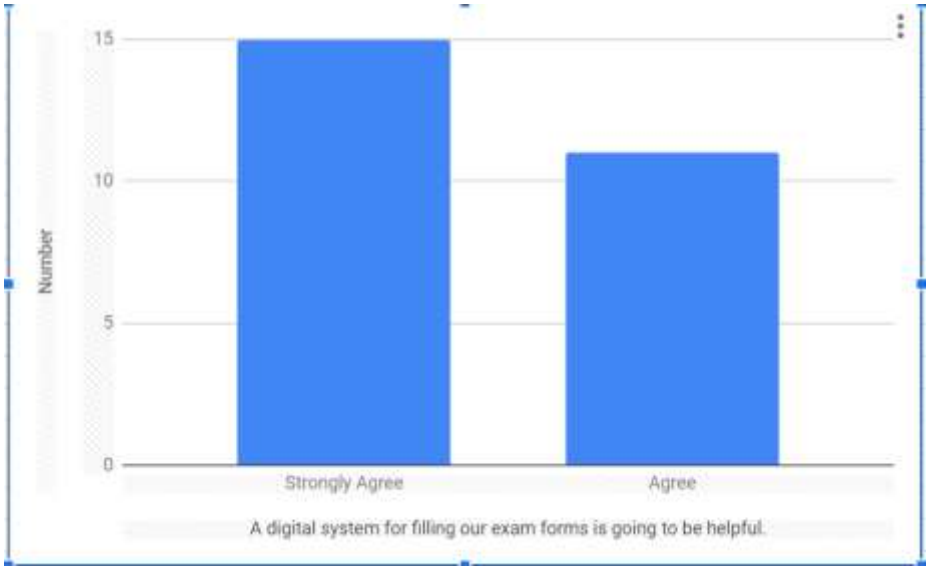


Figure 2 Survey result 2

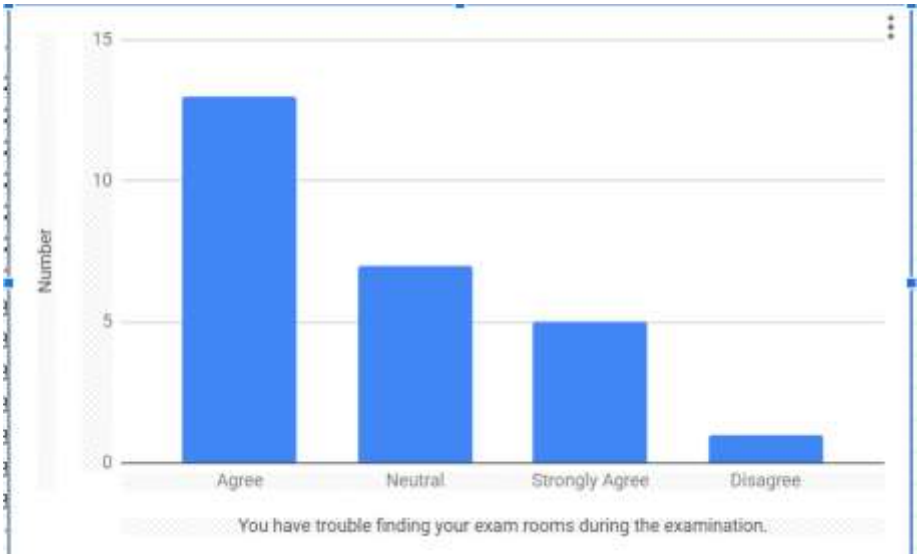


Figure 3 Survey result 3

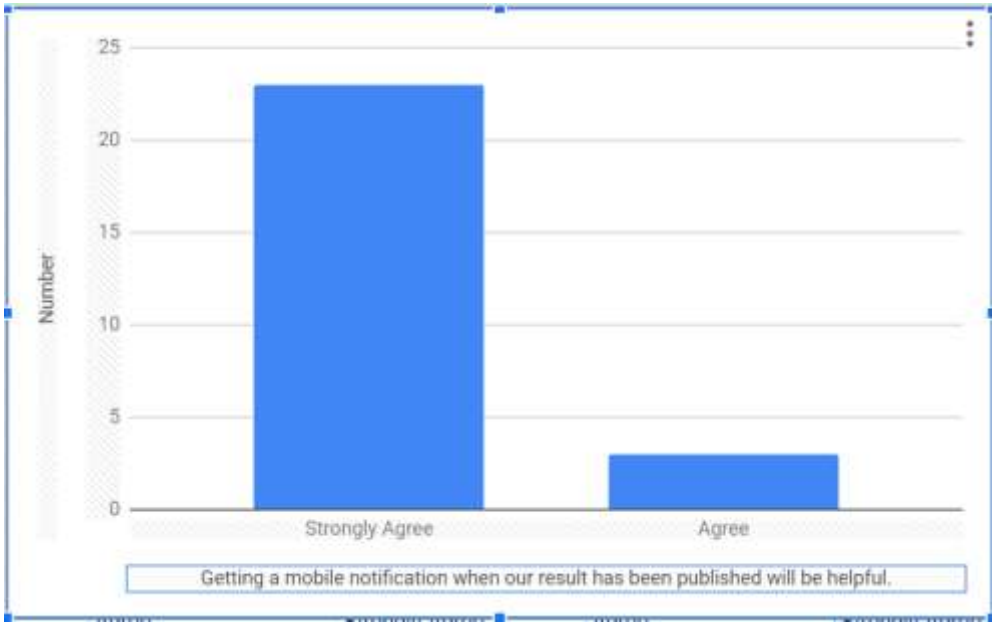


Figure 4 Survey result 4

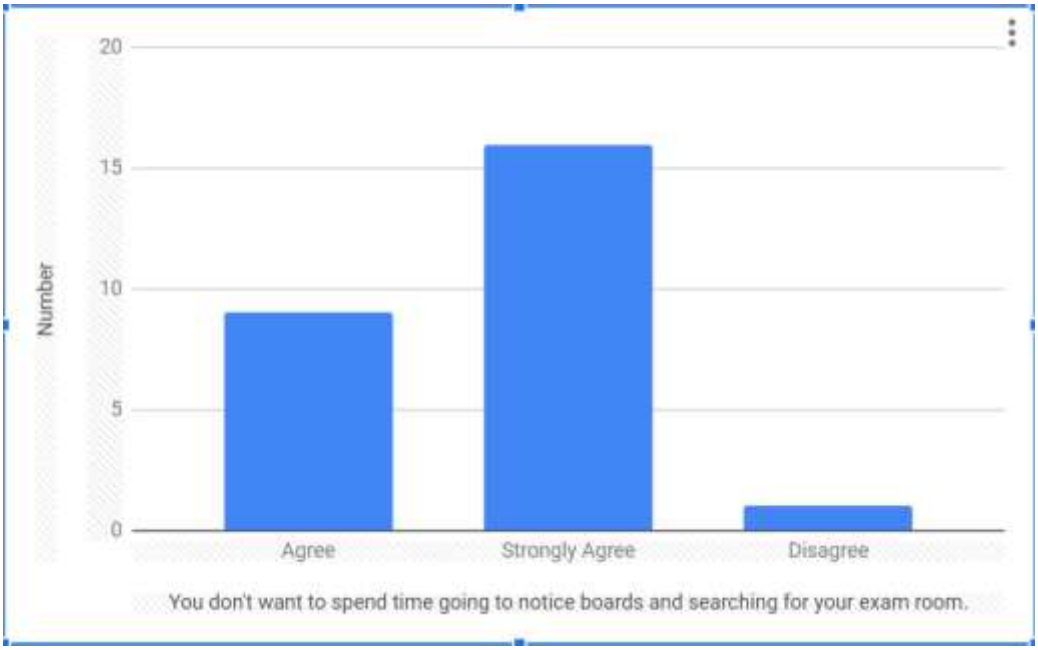


Figure 5 Survey result 5

2. Related Works

1. Notification system

This is a simple application that helps us generate and receive push notifications or cloud messaging to android devices using Firebase. We can receive notifications regarding different events through this app.

Features of this app:

- Notification generation and sending for different user classes
- Instantly receive the generated push notification

2. Firebase database program

Flutter blog app is an online blog app developed using Firebase real time database and Firebase authentication for user sign up and log in. In this flutter iOS app and flutter Android app, a user will be able to upload new blogs or posts and other users can see and watch blogs or posts of different users in this app.

Features of this app:

- Flutter firebase sign in using Firebase Authentication
- Flutter firebase sign out using Firebase Authentication
- Allow users to Add, Update, Delete New Posts using Firebase Real Time Database
- Fetch users posts from Firebase database and display it on home page

3. Design and Implementation

The sequential procedures that we performed during the project are as follows:

Feasibility Study:

While doing feasibility study different technical, economical, operational, schedule feasibility were considered. An important issue for the development of the project was the selection of suitable front-end and back-end. When we decided to develop the project, we went through an extensive study to determine the most suitable platform that suits the needs of the organization as well as helps in development of the project. Hence, we selected Android the front end tool and Firebase as back end tools for developing our project.

The Exam section Management System does not require enormous amount of money to be developed hence this project was planned economically.

The system is also operationally feasible as it is very easy for the End users to operate it. It only needs basic information about Windows platform. And the project was developed in the allocated amount of time.

System Analysis:

The study conducted during the analysis phase is largely based on feasibility study. The system is effectively used to provide form filling facility and all the notices are digitalized.

System Design:

In this software we have used different dummy modules like admin and students. The admin module handles the entire system. They are able to add courses, publish notices and manage seat plans for examination. The student module can maintain their personal profile, fill exam forms and view their online GPA.

Flowcharts:

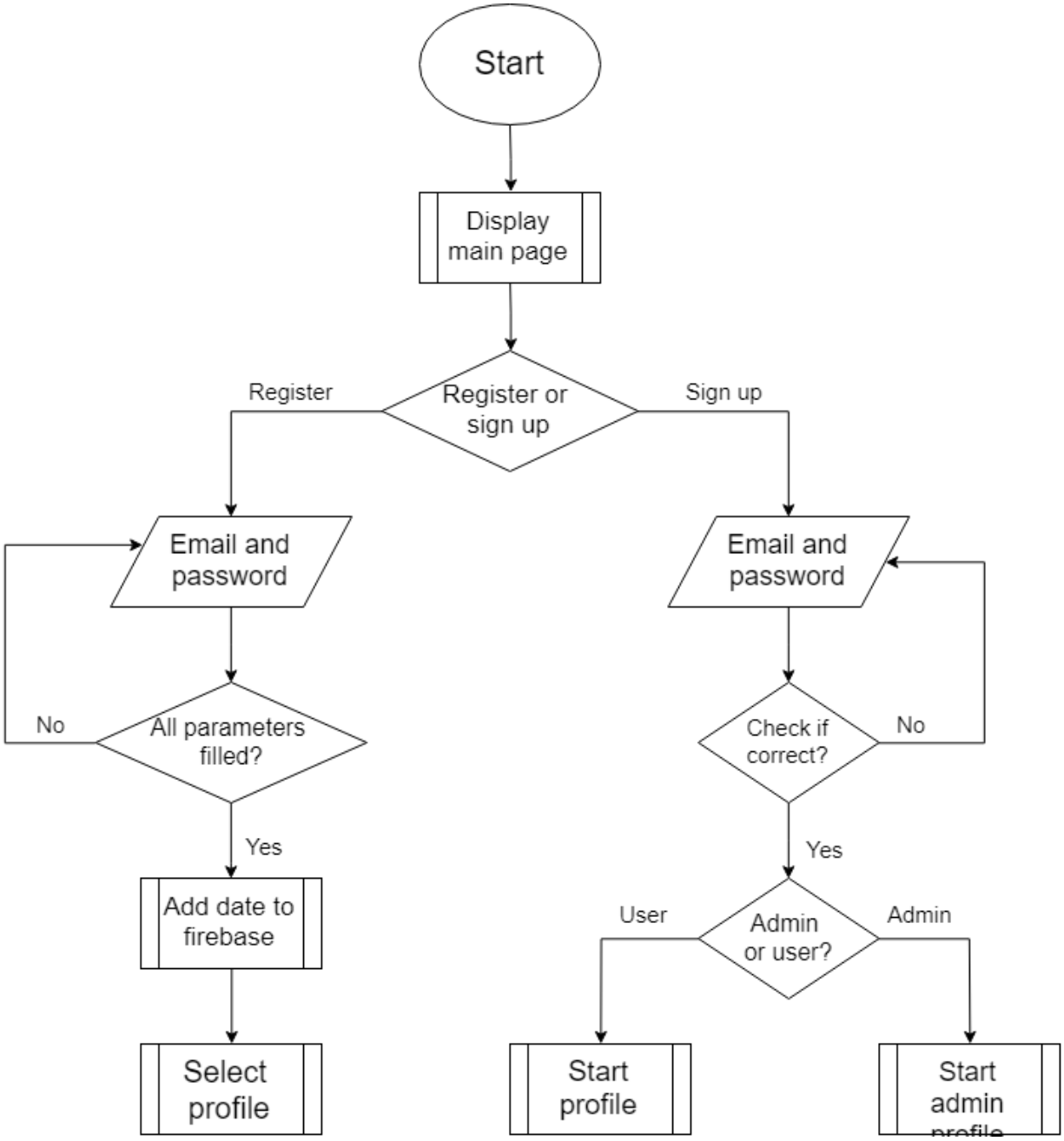


Figure 6: Main Activity

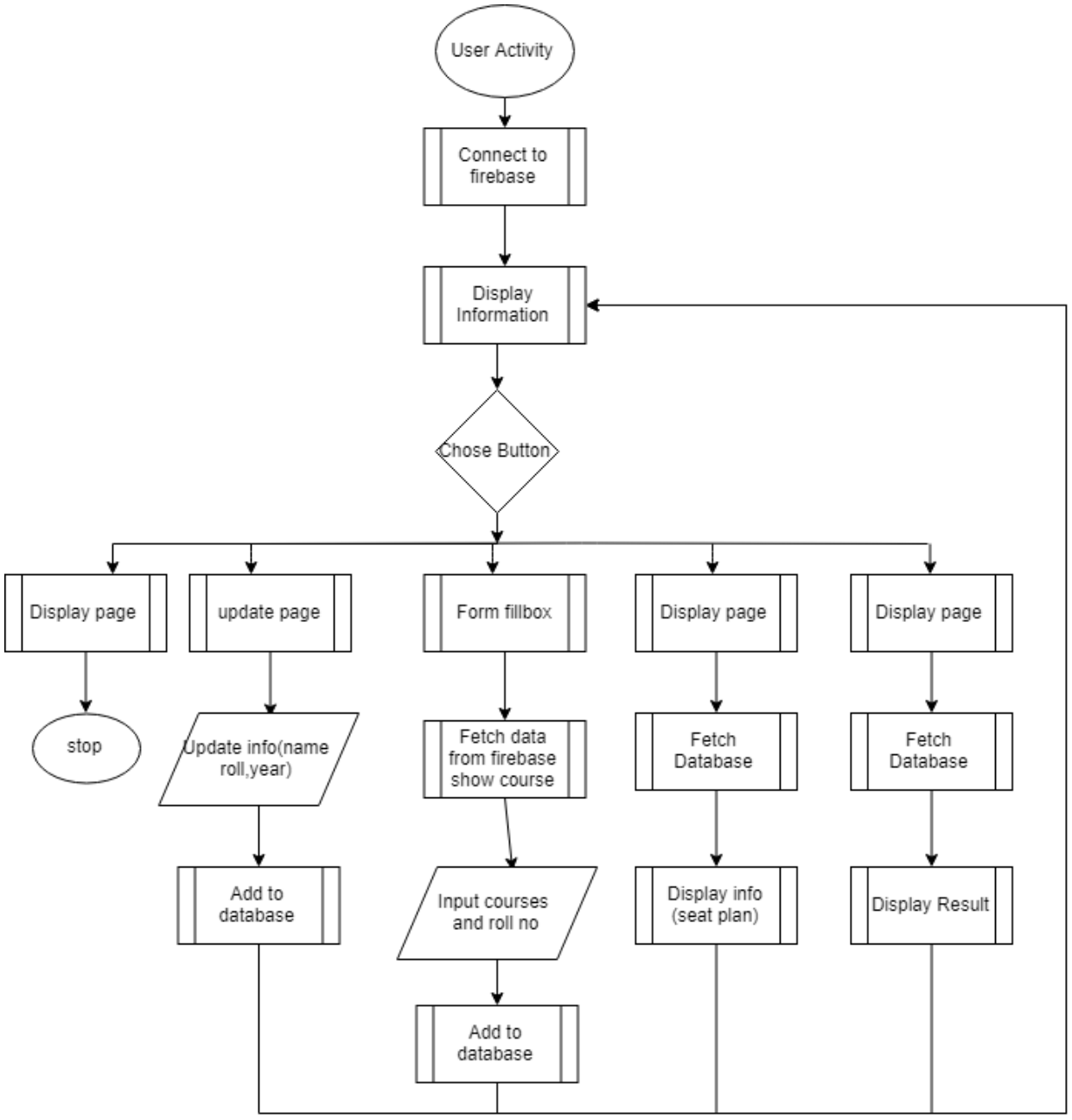


Figure 7: User Activity

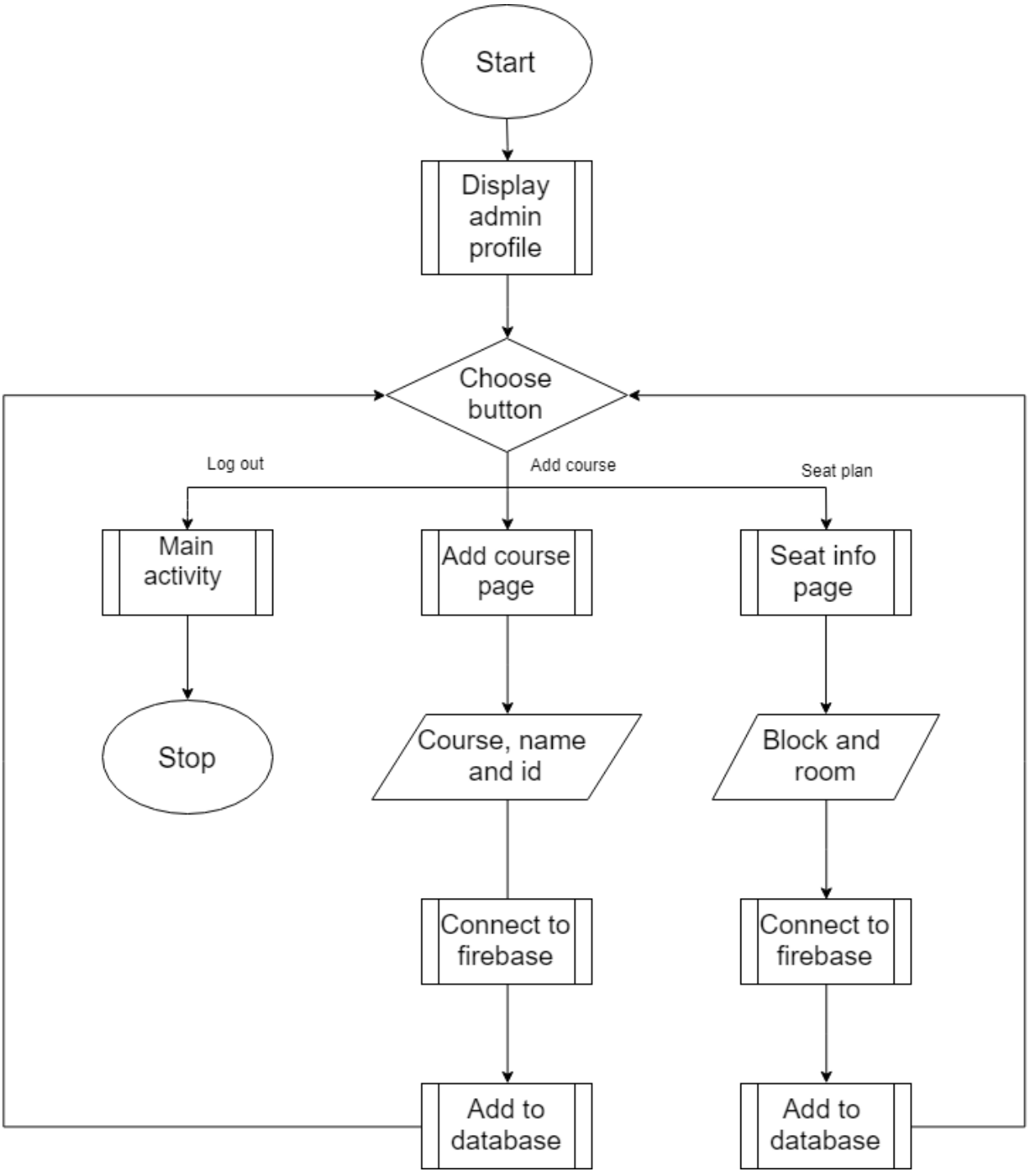


Figure 8: Admin Activity

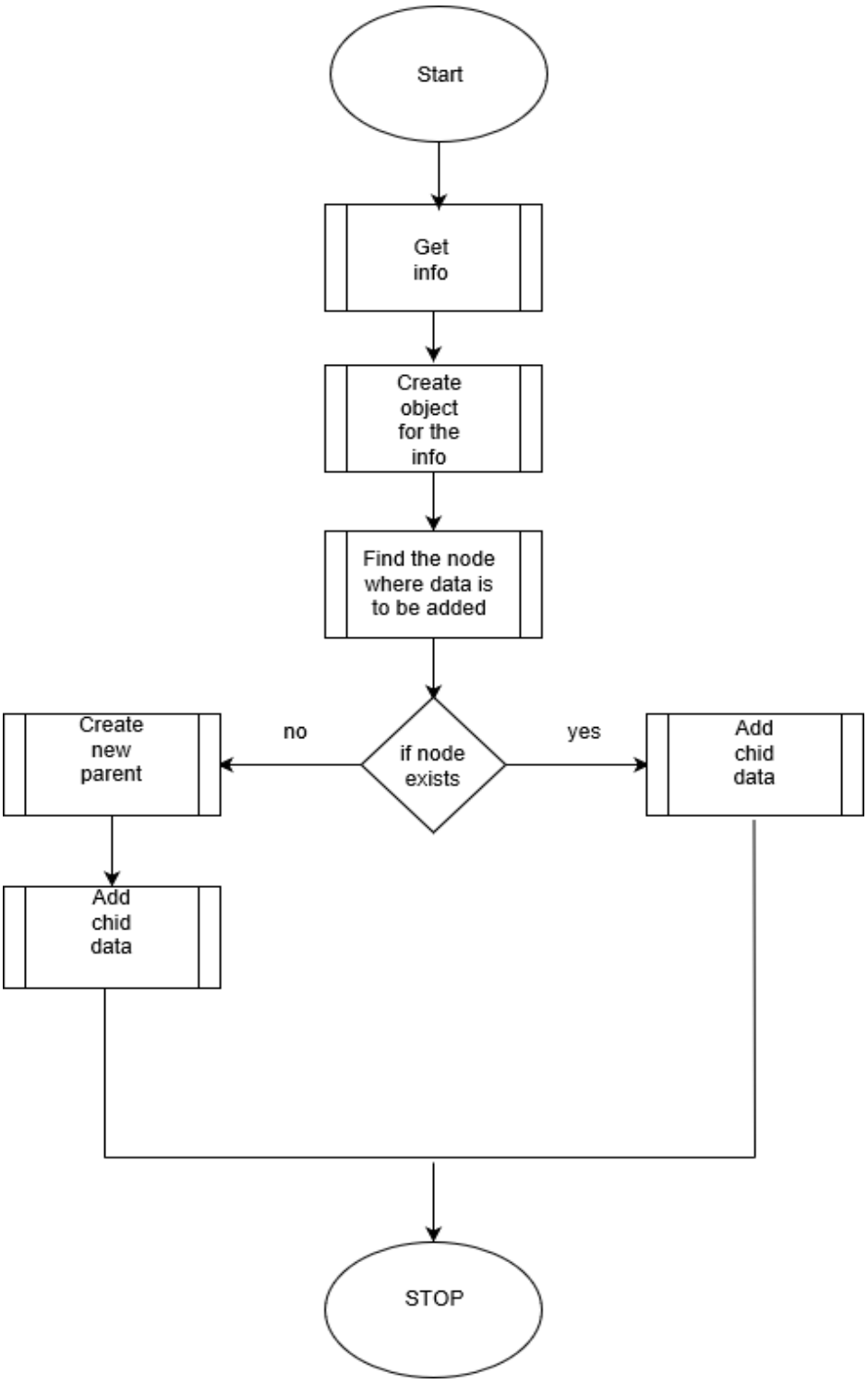


Figure 9: Add to Firebase

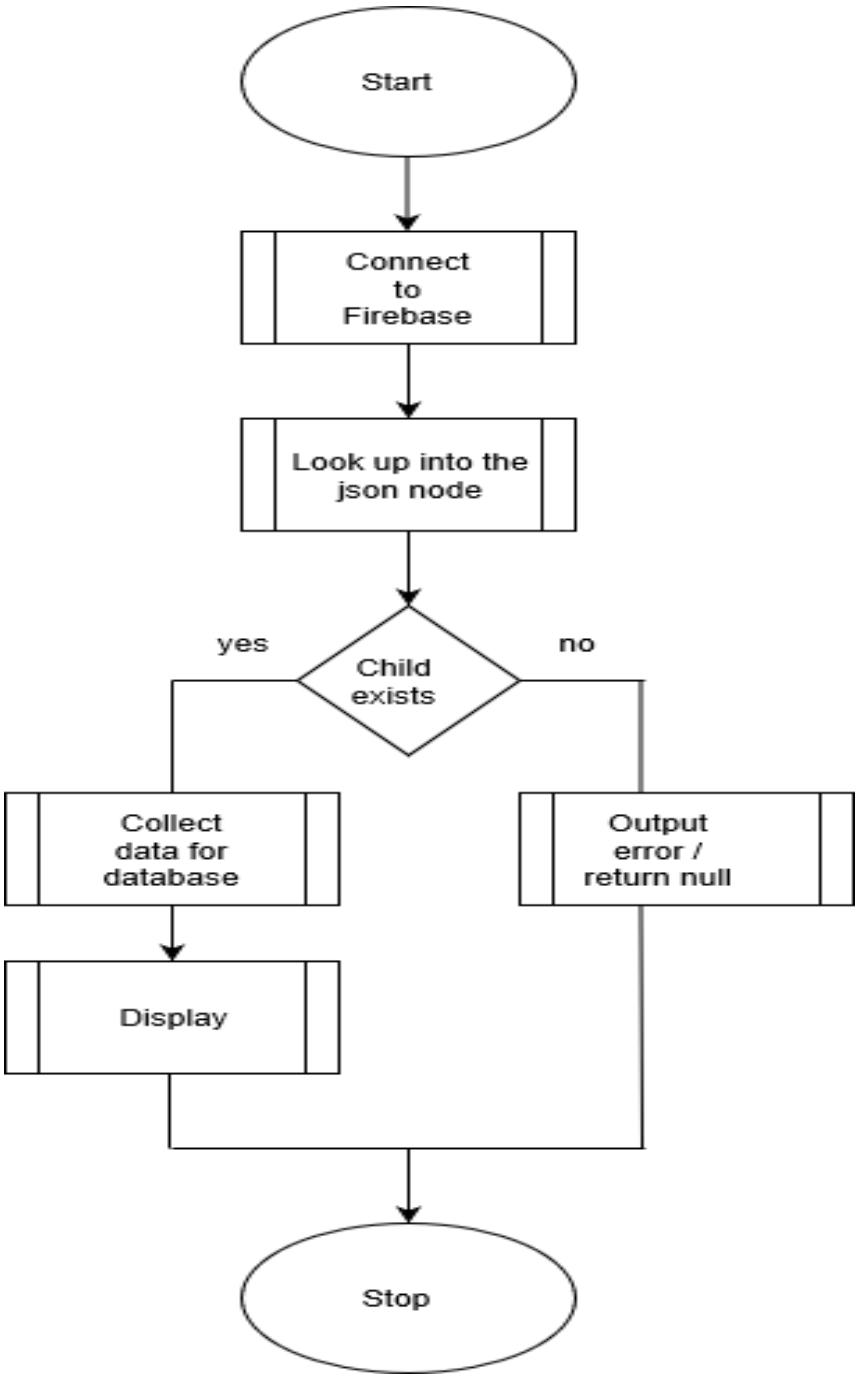


Figure 10: Retrieve data

Coding:

We have completed our project using Android as the core language and have used Firebase for the database

Testing:

Testing was performed using the program test and system test. In program test after the completion of the code, the system was tested using dummy data and bugs were removed. In system test the system was tested on actual data by carrying out a survey.

Implementation:

The application is hosted live in the domain and only the admin has the authority to change, update the contents of the application and monitor it's working.

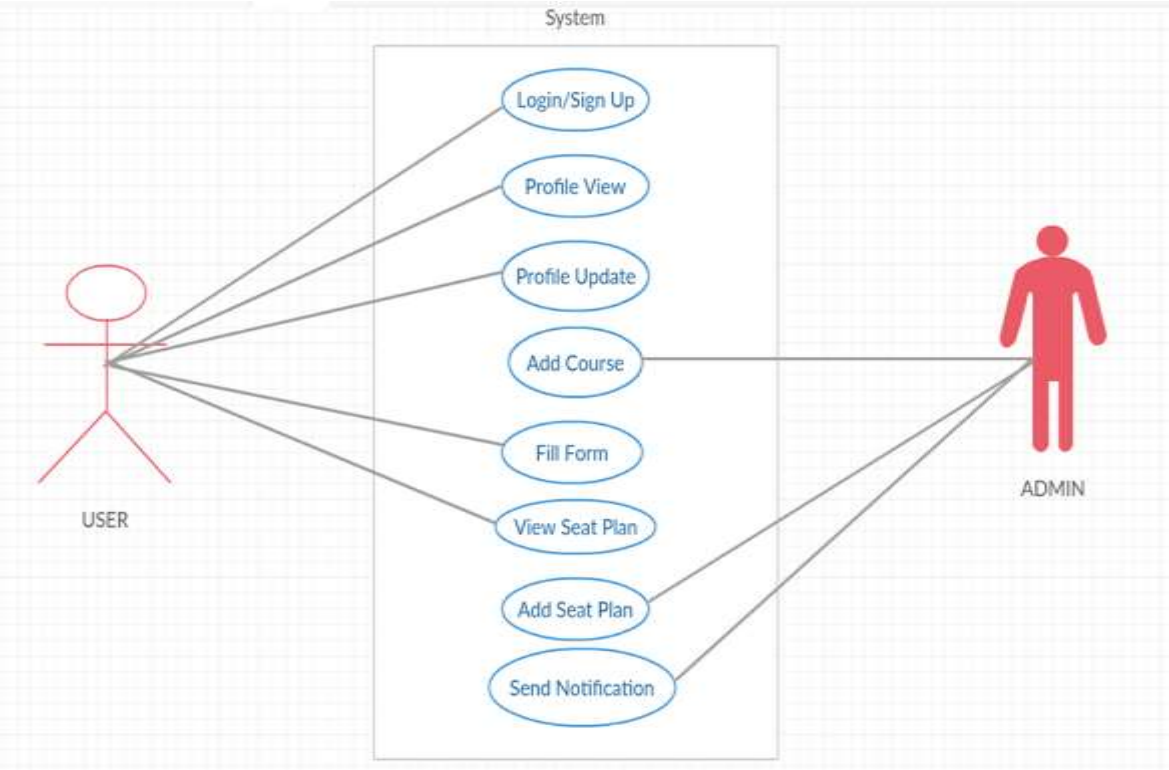


Figure 11: Use Case Diagram

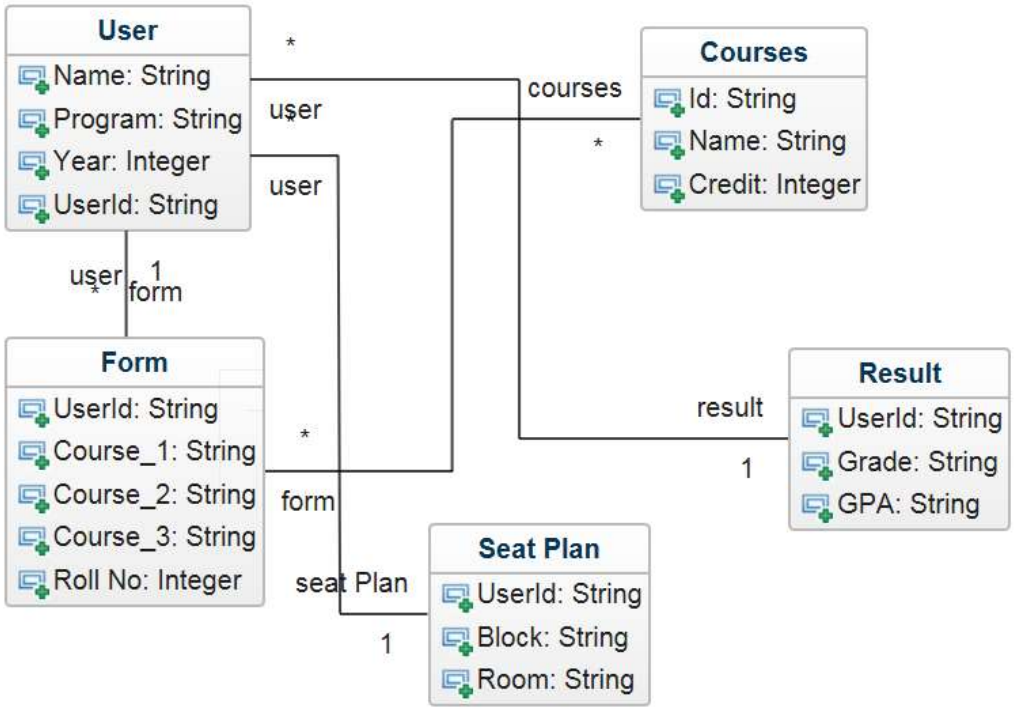


Figure 12: Database

3.1 System Requirements

3.1.1 Software Specification:

Our project Exam Section Management System uses Android OS and devices.

System requirements
1. For operating system
Minimum required Android version: Android KitKat 4.0
Maximum Android version supported: Android Pie 9.0

2. For device
Minimum required SoC: Snapdragon 450
Minimum required RAM: 1 GB

3.1.1.1 Front end Tools:

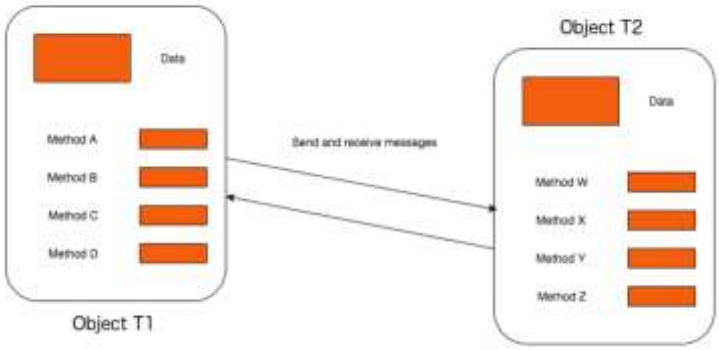
Android Studio is the official Integrated Development Environment (IDE) for Android app development. Android Studio offers even more features that enhance your productivity when building Android apps. A flexible Gradle-based build system, a fast and feature-rich emulator, a unified environment where you can develop for all Android devices, testing tools and frameworks, Lint tools to catch performance, usability, version compatibility, and other problems, C++ and NDK support, built-in support for google cloud platform, making it easy to integrate Google Cloud Messaging and App Engine

3.1.1.2 Backend Tools:

Object-oriented programming (or OOP) is a paradigm or pattern of programming whereby the solution to a programming problem is modelled as a collection of collaborating objects. Objects collaborate by sending messages to each other. It is most suitable for managing large, complex problems.

An object is an entity that possesses both state (or properties or attributes) and behavior.

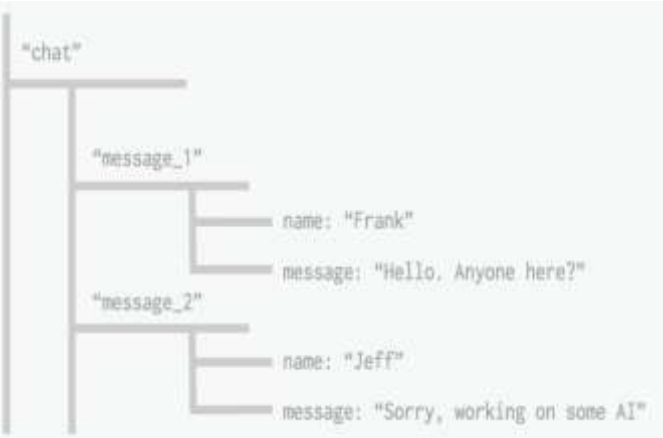
A class is a special kind of object that's used as a template for creating instances of itself. Think of it like a cookie cutter that produces cookies (or objects).



Java is a battle-tested language that has proven its speed and reliability over the last 20 years. Java is one of the most popular JVM languages. Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc. It is one of the most popular programming language in the world. It is easy to learn and simple to use. It is open-source and free. It is secure, fast and powerful. It has a huge community support (tens of millions of developers).

Firebase Cloud Messaging (FCM) provides a reliable and battery-efficient connection between your server and devices that allows you to deliver and receive messages and notifications on iOS, Android, and the web at no cost. The nice thing about using FCM is that as long as you are not sending an extensive amount of messages, more than 1000 at a clip and a typical delivery time of about 250ms with a payload of less than 4K and a persistence limit of 100 messages prior to messages collapsing, its free and it works.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync between your users in realtime. The Realtime Database is really just one big JSON object that the developers can manage in realtime.



With just a single API, the Firebase database provides your app with both the current value of the data and any updates to that data. Realtime syncing makes it easy for your users to access their data from any device, be it web or mobile. Realtime Database also helps your users collaborate with one another.

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more.Firebase Authentication integrates tightly with other Firebase services, and it leverages industry standards, so it can be easily integrated with your custom backend. The FirebaseUI Auth component implements best practices for authentication on mobile devices and websites, which can maximize sign-in and sign-up conversion for your app. It also handles edge cases like account recovery and account linking that can be security sensitive and error-prone to handle correctly.

4. Discussion on the Achievements

Firstly we decided that an android mobile application would be the best choice for our users because, all the student will be using their mobile phones and they carry their phones all the time. Sending notification to the students in their mobile would help solve one of the main objectives of the project and that was to let students know about various updates regarding the examination. After this the main problem that was faced during the project was to create an online database. As our project was wholly based in retrieving and as well as sending data to an online source which will store the user information too. Thus we made use of the Firebase feature provided by Google which can basically help create a database for storing the information and also provide us with the tools for online real-time notifications system.

Methodology

Our first task was to setup the android environment in our android studios. Then we declared various dependencies that will be used in our program. The dependencies we have used are for firebase core, firebase authentication, firebase real-time database and firebase messaging.

The dependencies help us use various types of functions that these dependencies have in-built which makes our work very easy.

The Dependencies are:

implementation 'com.google.firebase:firebase-database:16.0.1'

implementation 'com.google.firebase:firebase-core:16.0.1'

implementation 'com.google.firebase:firebase-auth:16.0.1'

implementation 'com.google.firebase:firebase-messaging:17.1.0'

Activities

Then our activity classes were set up and we created the home screen for the project. The activities represent the main process that are carried out in our project and function in these activities help to navigate between the pages. The various activities and their work in our program are discussed below.

MainActivity

This is the main activity of our program and it initializes firebase and connects our program with the online database. The next thing done in this activity is it defines the online cloud messaging which will help send notifications in our application to the users. Then the program asks the user to register the account or to login with their registered account.

The other activities used are discussed briefly below:

LoginActivity

Login for the user. It asks for password and email to the user and if they are registered users of the program, displays the profile page of the user. Also if the user is admin it redirects the admin to the admin page

Activities for the user

FormFill Activity – Fill the form choosing the available courses.

ViewSeat Activity – See the form as well as the seat plan.

View Result Activity - See the GPA.

UpdateProfile Activity- Help add or change the user information.

Activities for the admin

AddCourse Activity – Enable admin to add a new course to the database

SeatPlan Activity – Enables admin to change the seat plan for the users.

Java Classes

Java classes have been very useful to create objects and add the object into the database instead of adding each and every information without making an object. There are four main java classes that are used:

UserInfo

It help create an object with user information, name, program and year.

FormInfo

It help create an object with form info containing roll, and courses.

SeatInfo

It contains block and room where the examination will be held.

Major Functions

Intent –

An intent is an abstract description of an operation to be performed. It can be used with startActivity to launch an Activity, broadcastIntent to send it to any interested BroadcastReceiver components, and Context.startService(Intent) or Context.bindService(Intent, ServiceConnection, int) to communicate with a background Service.

FirebaseAuth

public abstract class FirebaseAuth extends Object

The entry point of the Firebase Authentication SDK.

First, obtain an instance of this class by calling getInstance().

Then, sign up or sign in a user with one of the following methods:

createUserWithEmailAndPassword(String, String)

signInWithEmailAndPassword(String, String)

signInWithCredential(AuthCredential)

Finally, call getCurrentUser() to get a FirebaseUser object, which contains information about the signed-in user.

DatabaseReference

public class DatabaseReference extends Query

A Firebase reference represents a particular location in your Database and can be used for reading or writing data to that Database location.

This class is the starting point for all Database operations. After you've initialized it with a URL, you can use it to read data, write data, and to create new DatabaseReferences.

Features:

- Online exam form filling and course selection.
- Notification service when result is published
- Digitalization of all the notices published by exam section
- User Profile Maintenance
- Online mark-sheet.

Screenshots

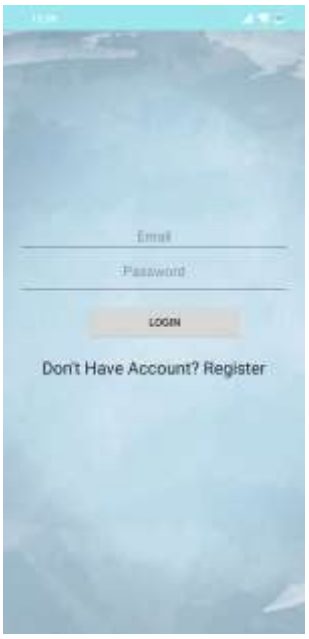


Fig. Register

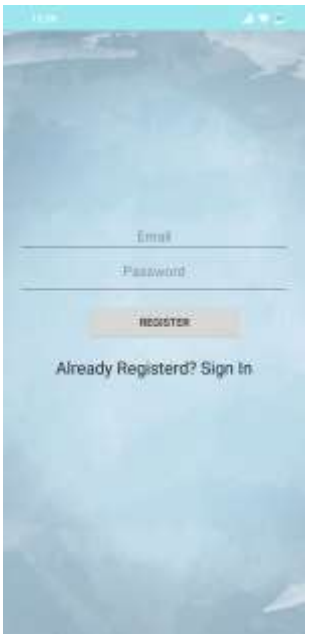


Fig. Login

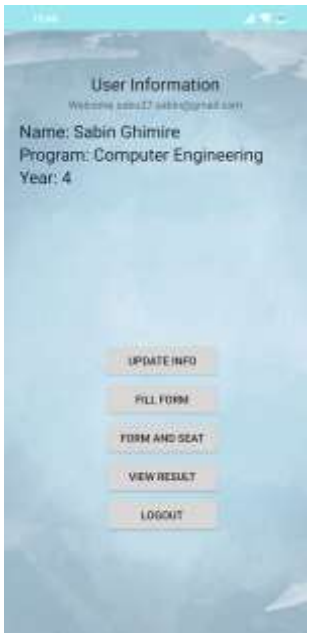


Fig. User Profile

Fig. Update Info

Fig. Form Fill

Fig. Seat Plan

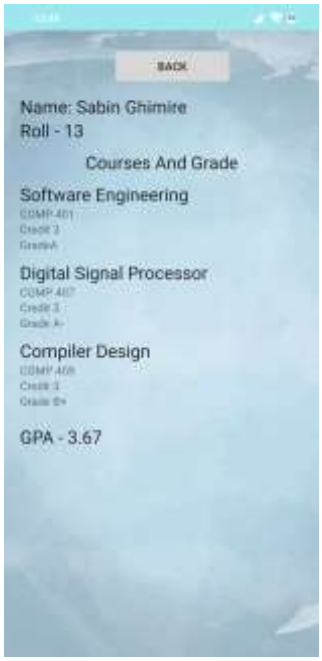


Fig. Result

Fig. Admin

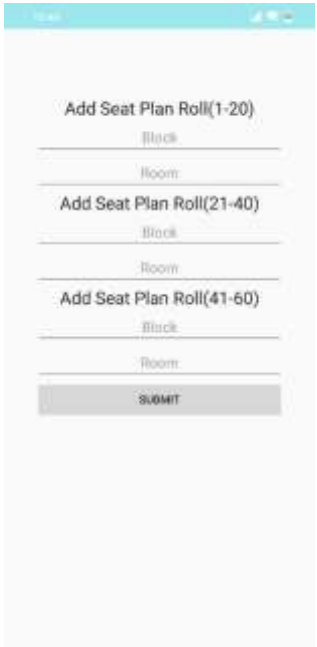


Fig Add Seat Plan

Fig Add Course

5. Conclusion and Recommendation

The project “Examination Section Management System” is for computerizing the daily works carried out by exam section. The application takes care of all the requirements of our exam section and is capable of providing easy and effective digital information related to exams, online exam form filling, shows GPA and lets student to maintain their personal profile.

5.1 Limitation

- Firebase console has to be used so not all activities can be controlled with mobile phone.
- Front end can be made better.
- GPA generator is not our system so data has to be entered manaually.

5.2 Future Enhancement

- GPA calculator can be added.
- Implementing across all platforms.

Reference

[1] Tutorials Point. (2018, July 21). SDLC Overview. Retrieved July 29, 2018, https://www.tutorialspoint.com/sdlc/sdlc_overview.htm

[2] Basic android developing1 (Last accessed: 1st June, 2017): https://www.developer.android.com

[3] Java Programming2 (Last accessed: 13st July, 2017): http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

[4] UI development3 (Last accessed: 3rd August, 2017):

https://www.udacity.com/course/viewer#!/c-ud837/l-4034888704/e-4221008776/m-4220918672

Appendices

Table 1: Gantt Chart:

Work / Week	1.	2.	3.	4.	5.	6.	7.	8.
Research and study								
Graphics Designing								
Core Programming								
Testing								
Documentation								

INDEX

Task completed